

GBASE

GBase 8c 技术白皮书



GBase 8c 技术白皮书，南大通用数据技术股份有限公司

GBase 版权所有©2022，保留所有权利

版权声明

本文档所涉及的软件著作权及其他知识产权已依法进行了相关注册、登记，由南大通用数据技术股份有限公司合法拥有，受《中华人民共和国著作权法》、《计算机软件保护条例》、《知识产权保护条例》和相关国际版权条约、法律、法规以及其它知识产权法律和条约的保护。未经授权许可，不得非法使用。

免责声明

本文档包含的南大通用数据技术股份有限公司的版权信息由南大通用数据技术股份有限公司合法拥有，受法律的保护，南大通用数据技术股份有限公司对本文档可能涉及到的非南大通用数据技术股份有限公司的信息不承担任何责任。在法律允许的范围内，您可以查阅，并仅能够在《中华人民共和国著作权法》规定的合法范围内复制和打印本文档。任何单位和个人未经南大通用数据技术股份有限公司书面授权许可，不得使用、修改、再发布本文档的任何部分和内容，否则将视为侵权，南大通用数据技术股份有限公司具有依法追究其责任的权利。

本文档中包含的信息如有更新，恕不另行通知。您对本文档的任何问题，可直接向南大通用数据技术股份有限公司告知或查询。

未经本公司明确授予的任何权利均予保留。

通讯方式

南大通用数据技术股份有限公司

天津市高新区华苑产业园区开华道22号普天创新园东塔20-23层(300384)

电话：400-013-9696

邮箱：info@gbase.cn

商标声明

GBASE[®] 是南大通用数据技术股份有限公司向中华人民共和国国家商标局申请注册的注册商标，注册商标专用权由南大通用数据技术股份有限公司合法拥有，受法律保护。未经南大通用数据技术股份有限公司书面许可，任何单位及个人不得以任何方式或理由对该商标的任何部分进行使用、复制、修改、传播、抄录或与其它产品捆绑使用销售。凡侵犯南大通用数据技术股份有限公司商标权的，南大通用数据技术股份有限公司将依法追究其法律责任。

目 录

目 录	II
1 产品简介	4
1.1 发展历程	4
1.2 产品定位	4
1.3 应用场景	5
1.3.1 典型应用场景举例	6
1.4 技术特点	7
1.5 功能简介	10
2 产品家族	12
2.1 GBase 8c 数据库	12
2.1.1 单机/主备架构	12
2.1.2 分布式架构	13
2.2 通用管理平台	14
2.3 数据迁移平台	15
3 环境及技术指标	17
3.1 软硬件运行环境	17
3.2 技术指标	17
4 核心技术	19
4.1 自适应事务处理机制	19
4.2 数据分布策略	19
4.3 高性能	20
4.3.1 NUMA 优化	20
4.3.2 并行技术	20
4.3.3 原位更新	20

4.3.4	算子下推	21
4.3.4.1	单表查询下推	21
4.3.4.2	Join 查询下推	23
4.4	高可用	24
4.4.1	主备式高可用架构	24
4.4.2	分布式高可用架构	25
4.5	分布式事务	29
4.6	全局死锁解除	31
4.7	事务状态保持	33
4.8	全局 CDC	33
4.9	备份恢复	33
4.10	读写分离	34
5	产品高级特性	35
5.1	多租户	35
5.2	灰度发布	35
5.3	强一致性分布式事务	36
5.4	多部署方式	36
5.5	安全特性	37
5.6	空间数据库	38
6	开发接口	39
6.1	ODBC	39
6.2	JDBC	39
6.3	ADO.NET	39
6.4	C API	40
6.5	Python API	40

1 产品简介

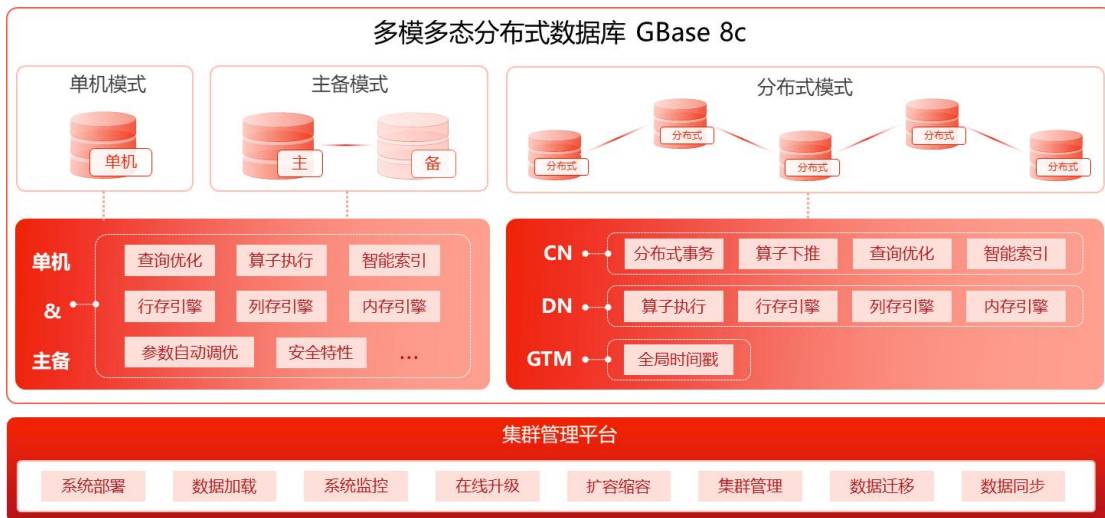
1.1 发展历程



南大通用自 2010 年起涉足分布式数据库领域，厚积薄发，十年磨一剑，推出基于 openGauss3.0 版本的多模多态分布式数据库 GBase 8c。

1.2 产品定位

GBase 8c 是基于 openGauss 构建的一款多模多态的分布式数据库，支持行存、列存、内存等多种存储模式和单机、主备与分布式等多种部署形态。GBase 8c 具备高性能、高可用、弹性伸缩、高安全性等特性，可以部署在物理机、虚拟机、容器、私有云和公有云，为关键行业核心系统、互联网业务系统和政企业务系统提供安全、稳定、可靠的数据存储和管理服务。



- ✓ **多模多态**: GBase 8c 支持行存、列存、内存三种存储模式; 支持单机、主备与分布式三种部署形态, 满足用户各种业务/场景需求。
- ✓ **极高性能**: GBase 8c 依托 openGauss 内核多核架构的并发控制技术, 结合鲲鹏硬件优化, 并在内核关键结构上采用了 Numa-Aware 的数据结构, 同时依托分布式扩展能力, 具备高达 0.9 以上的弹性扩展能力, 提供集群极高性能能力。
- ✓ **极高可用**: GBase 8c 主备式形态提供主备同步、异步以及级联备机多种部署方式; 分布式形态集群全局无单点, 高可用可达到 99.999%; 同时分布式集群支持两地三中心, 支持异地多活部署, 业务保持连续在线, 数据永不丢失。
- ✓ **极致弹性**: GBase 8c 支持在线扩缩容, 计算存储分离的架构能够保证集群计算和存储能力随节点数的增加可以线性提升。
- ✓ **极致安全**: GBase 8c 支持全密态计算能力、支持访问控制、加密认证、数据库审计、动态脱敏等安全特性, 提供全方位端到端的数据安全保护。
- ✓ **高兼容性**: GBase 8c 支持标准 SQL 语法, 完全兼容 PostgreSQL 语法, 高度兼容 Oracle 语法。
- ✓ **分布式能力**: GBase 8c 支持强一致性的分布式事务, 支持算子下推, 具备多线程并行业务处理能力, 采用 MVCC 技术实现读写并发处理能力。

1.3 应用场景

● 关键行业核心业务

关键行业核心系统对数据库的性能、稳定性和安全性有着严苛的要求。随着关键行业核心系统业务量的不断增长, 传统的集中式数据库对高并发、高吞吐量的需求支持能力不足, 分布式数据库成为解决这一问题的有力武器。

- ✓ GBase 8c 具备分布式事务强一致性、计算能力和存储能力线性扩展的核心能力;
- ✓ 同时支持多地多中心和异地多活, 提供 99.999%的高可用性;
- ✓ 数据库配套可视化智能运维平台, 极大降低客户运维复杂度。

● 互联网业务

随着互联网业务，尤其是移动互联网业务的飞速发展，互联网应用对海量数据下数据库的并发量、吞吐量等性能要求越来越高。并且，为了支撑秒杀等特定场景的极端性能需求，自动伸缩也成为互联网应用的必要需求。

- ✓ GBase 8c 的弹性伸缩能力，可以根据业务需求进行在线扩容缩容，完美应对极端业务场景的性能需求。
- ✓ 灰度发布的特性，为互联网业务频繁上下线提供了有力支撑。

● 政企业务上云

随着云计算和虚拟化技术的不断深入发展，政企业务上云成为主流的架构选择。

- ✓ GBase 8c 支持物理机、容器、私有云、公有云多种部署方式，为政企业务上云提供了灵活的选择。
- ✓ 支持支持行存、列存、内存三种存储模式，支持单机、主备与分布式三种部署形态，满足用户各种业务需求。
- ✓ GBase 8c 支持数据传输加密、存储加密，对云上数据安全提供了有力的保证。

1.3.1 典型应用场景举例

应用场景	场景特点	优点
金融行业 - 信用卡账单查询 - 交易明细查询	1、并发量相对比较高 2、主要是读操作，较少或不涉及分布式事务	基本上没有跨节点写操作，可以发挥分布式的优势，通过多节点并行处理，将整个查询压力平均分散到所有节点，从而充分来释放整个集群的性能
电信行业 - 用户账单查询 - 通话详单查询		
政务业务 - 核酸结果查询 - 健康码扫码亮码		
支付类业务 - 信用卡记账 - 话费详单		
传统行业业务	1、数据量大 2、业务要求数据强一致性	数据量大，需要进行数据拆分，适合分布式部署；

		要求数据强一致性，常见的分库分表、分布式中间件方案难以满足，适合原生分布式架构产品；
访问量波动明显 - 全国性集中填报与提交	业务具有定期波动性，应用不易处理	利用分布式架构弹性伸缩特性，忙时对系统计算节点进行扩容，增加整体的并发访问处理能力；闲时减少节点数量，节省资源，降低运营维护成本
GIS 平台对接或数据处理	支持 GIS 特性，包括地理空间数据的存取功能，提供地理信息的快速检索功能等	openGauss/pg 系产品对 GIS 天然支持比较好，且 GBase 8c 作为分布式数据库产品在构建“GIS 云平台”相关场景下可以提供良好的底层支持

1.4 技术特点

作为一款多模多态分布式数据库产品，GBase 8c 具有强一致性的全局事务、计算存储分离、灵活的数据分布、灵活的部署方式、多模多态、在线扩容缩容、在线升级、数据高可用、高安全性、异地多活、数据高效加载、集群备份恢复、易维护、标准化、兼容国产生态等技术特点。

- **自动容错的强一致性全局事务**

GBase 8c 采用两阶段提交协议和全局时间戳来保证全局事务的强一致性，每个跨节点的事务，要么全部成功，要么全部失败，不会出现某些节点事务成功，另外一些节点事务失败的情况，实现全局事务的强一致性。GBase 8c 的事务处理具有自动容错能力，某个正在处理事务的节点发生故障后，新的节点会继续进行未完成的事务处理，不需要应用程序重新请求。

- **计算存储分离**

GBase 8c 采用 share nothing 架构，计算和存储分离。可以根据业务需求，对计算能力和存储能力分别进行水平扩展，达到降低总体拥有成本的目的。

- **灵活的数据分布**

用户可以按照业务场景的需要，选择数据分布策略，从而在性能、可靠性和灵活性间获得最佳匹配。

GBase 8c 支持复制表和分布表。复制表用于存储只读或者读多写少的数据，可以在本地和分布表进行联合查询，从而大幅提升查询的性能。分布表用于存储单表规模较大的数据，通过 Hash 等方式分布到各个存储节点，降低单表数据量，提升数据读写性能。

● 灵活的部署方式

用户可以根据自身 IT 基础设施建设情况，结合性能、便捷、安全等需求的综合考量，选择将 GBase 8c 部署在不同的环境。GBase 8c 支持物理机部署、虚拟机部署、容器部署、私有云部署和公有云部署。

● 多模多态

GBase 8c 支持行存、列存、内存三种存储模式，支持单机、主备与分布式三种部署形态，满足用户各种业务需求。

✓ 多存储模式：GBase 8c 支持多种存储模式以满足不同场景的业务需求：

1. 行存储引擎：主要面向 OLTP 场景设计，例如订货、发货、银行交易系统；
2. 列存储引擎：主要面向 OLAP 场景设计，例如数据统计报表分析系统；
3. 内存引擎：主要面向极致性能场景设计，例如银行风控场景。

✓ 多部署形态：GBase 8c 通过多租户方式实现多种部署形态，可提供单机部署、主备部署及分布式部署三种部署形态，并通过统一运维管理平台来进行管理，分别面向企业核心交易和未来海量事务型场景，打造差异化竞争力。

1. 单机形态：GBase 8c 支持单机部署，可以直接在一台服务器上部署数据库，这种部署的优势是最低成本，且部署简洁。
2. 主备形态：GBase 8c 支持一主多备的部署方式，主备之间可以采用同步或异步备份方式。这种部署方式部署简洁、交付高效，适用于较低数据量、追求极致单机性能，且要求数据备份的场景。
3. 分布式形态：分布式模式，支持分布式全组件冗余的高可用，支持计算存储分离的部署。可以根据业务需求对计算和存储能力分别进行水平扩展，适用于大数据量高并发且追求数据高安全性的场景。

- **在线扩容缩容**

GBase 8c 支持在线扩容缩容，扩容缩容时，数据自动进行重分布，应用系统无感知。在线扩容缩容期间，不中断业务使用，保证系统的持续可用性。

- **数据高可用**

GBase 8c 通过冗余机制来保证集群的高可用特性，多个数据副本之间采用主从复制的方式来保证数据的一致性。GBase 8c 支持同步、异步和半同步复制模式，让用户可以在高可用和高性能之间寻求平衡。GBase 8c 支持故障的自动恢复，无需人工干预。

- **安全性**

GBase 8c 提供完善的用户、角色、权限控制策略，提高数据库集群的安全性；

支持详尽的审计日志，可配置灵活的审计策略，记录数据库中与数据库操作相关的所有日志，也可以通过图形化的监视工具实现审计管理；

支持多种加密策略：支持列级、表级，以及库级加密；

支持主流加密函数，如 AES、MD5、SHA1、SHA 等国际标准算法，以及 SM3、SM4 等国密算法。

- **异地多活**

GBase 8c 提供异地多活的部署方式，满足对高可用有严苛要求的业务需求。采用异地多活的高可用部署方式，可以达到 RPO=0，RT0 秒级的高可用等级，提供机房级容灾能力和城市级容灾能力。

- **数据高效加载**

GBase 8c 基于策略的数据加载模式，能够充分发挥所有节点的计算能力，保证数据在加载速度和访问性能之间的平衡。

- **易维护**

提供图形化的部署配置、运维监控、数据同步、备份恢复等运维功能，降低运维难度，提升运维效率。

- **标准化**

支持 SQL92、SQL99、SQL2003 ANSI/ISO 标准，支持 ODBC、JDBC、ADO.NET 等接口规范；支持 C API、Python API 等接口。

● 多种生态支持

GBase 8c 分布式交易型数据库具有丰富的周边生态支持：

- ✓ 支持强大的地理信息系统（GIS）。GBase 8c 支持 PostGis 插件，可以有效支持空间地理数据存储，作为空间数据库，高效管理空间数据、数据测量和几何拓扑分析。
- ✓ 支持非关系数据类型 JSON。GBase 8c 不仅仅是一个分布式交易型数据库系统，同时还支持非关系数据类型 JSON。
- ✓ 支持 Foreign Data Wrappers (FDW)。GBase 8c 通过 FDW 支持用户通过 SQL 访问数据库集群之外的数据。FDW 功能提供一套编程接口，用户可以进行插件式的二次开发，建立外部数据源和数据库间的数据通道。当前 GBase 8c 支持 oracle_fdw、mysql_fdw、postgres_fdw，以及非关系型数据库 redis_fdw、mongodb_fdw，大数据 hive_fdw、hdfs_fdw 等。通过 FDW，GBase 8c 能够访问已有的多个数据源的数据。
- ✓ 兼容国产生态。GBase 8c 兼容飞腾、鲲鹏、海光等国产 CPU；兼容麒麟、统信等国产操作系统；兼容基于浪潮、泰山、华三、曙光等国产 CPU 的服务器。

1.5 功能简介

功能	描述
结构化查询语言	支持标准的 SQL-92/SQL:1999/SQL:2003 规范，支持 GBK 和 UTF-8 等主流字符集； 支持 CREATE、ALTER、DROP 等 DDL 语法； 支持 SELECT、INSERT、UPDATE、DELETE、MERGE 等 DML 语法，支持单表，多表联合查询。
数据类型	boolean 布尔数据类型； integer、smallint、bigint、decimal、real 等数值数据类型； char、varchar 等字符数据类型； data、time、interval、timestamp 等日期类型； bytea、text 等大对象数据类型； point、line、path、circle 等地理位置类型。
数据库对象	提供了数据库、表、索引、视图、存储过程、触发器、自定义

	函数、同义词等常用数据库对象的创建、修改和删除操作，支持数据库用户/角色的创建、删除操作，以及用户权限的分配与回收。
函数	支持多种标准函数，包括控制流函数、字符串函数、数值函数、日期和时间函数、转换函数、位函数、加密函数、信息函数、辅助函数、聚集函数、OLAP 函数、正则表达式函数等；支持 C、PYTHON 语言用户自定义函数扩展。
事务特性	支持 ACID 特性的事务，支持 RC 和 RR 事务隔离级别；支持悲观锁、MVCC（多版本并发控制）、闪回。
分布式特性	支持数据分片； 读写水平扩展； 强一致性分布式事务。
智能管理平台	提供图形化的智能管理平台。
接口	符合并支持 ODBC、JDBC、ADO.NET 等接口规范；支持 C API、Python API 等接口。
安全管理	支持 SSL 安全网络连接、用户权限管理、密码管理、安全审计等功能，保证数据库在管理层、应用层、系统层和网络层的安全性。

2 产品家族

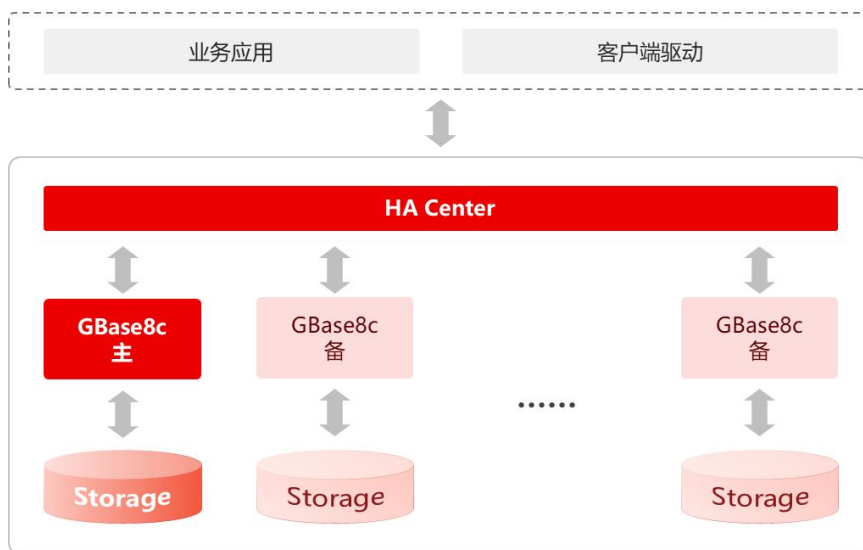
GBase 8c 的产品家族如下图所示，包括 GBase 8c 数据库、通用管理平台、数据迁移平台等。



2.1 GBase 8c 数据库

2.1.1 单机/主备架构

单机/主备形态的 GBase 8c 数据库，业务数据存储在一个物理节点上，主备节点之间通过同步或异步的方式进行高可用和扩展能力。

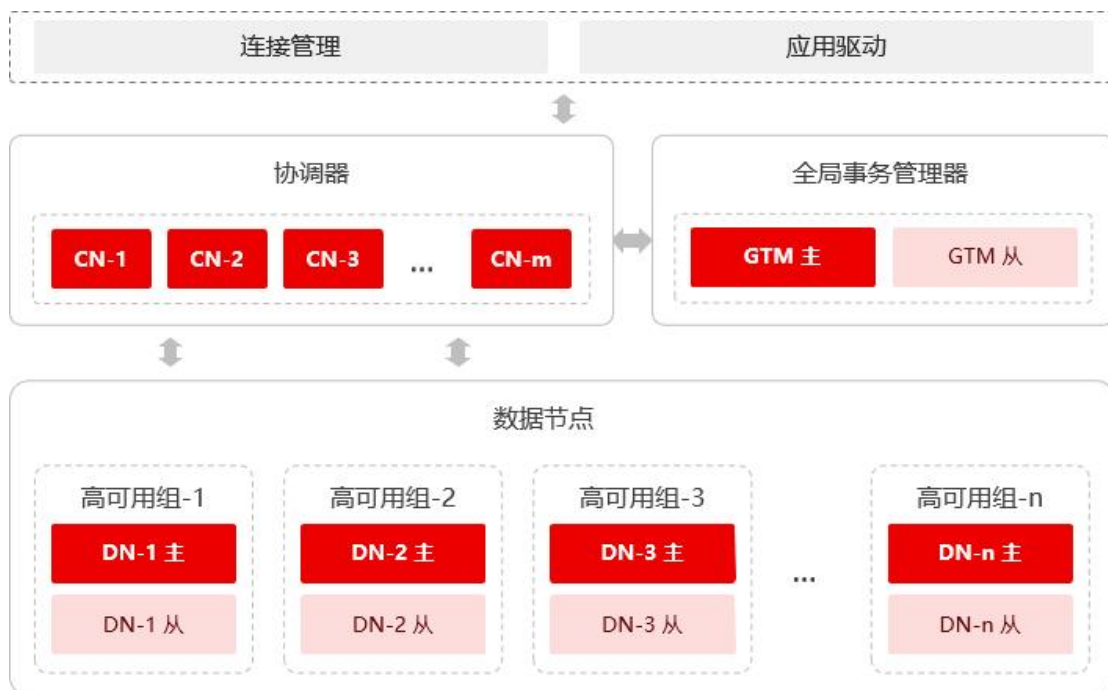


- ✓ **HA Center 高可用模块**: HA Center 是 GBase 8c 的高可用模块, 负责在故障情况下判断节点状态并进行状态切换。
- ✓ **GBase 8c 主 (备)**: 负责存储业务数据 (支持行存、列存、内存表存储)、执行数据查询及向客户端驱动返回执行结果。
- ✓ **Storage**: 服务器的本地存储资源, 负责持久化存储数据。

2.1.2 分布式架构

GBase 8c 数据库分布式形态采用 share nothing 的分布式架构, 计算节点和存储节点分离, 节点间通过高速网络进行通信, 所有节点都有主从互备, 确保系统的高可用性。

由于没有资源共享, 增加节点就可以线性地扩展集群的存储能力和计算能力, 满足业务规模增长的要求。



如上图所示, GBase 8c 的主要节点分为三类, 分别是协调器 (Coordinator, CN)、数据节点 (Data Node, DN)、全局事务管理器 (Global Transaction Manager, GTM)。

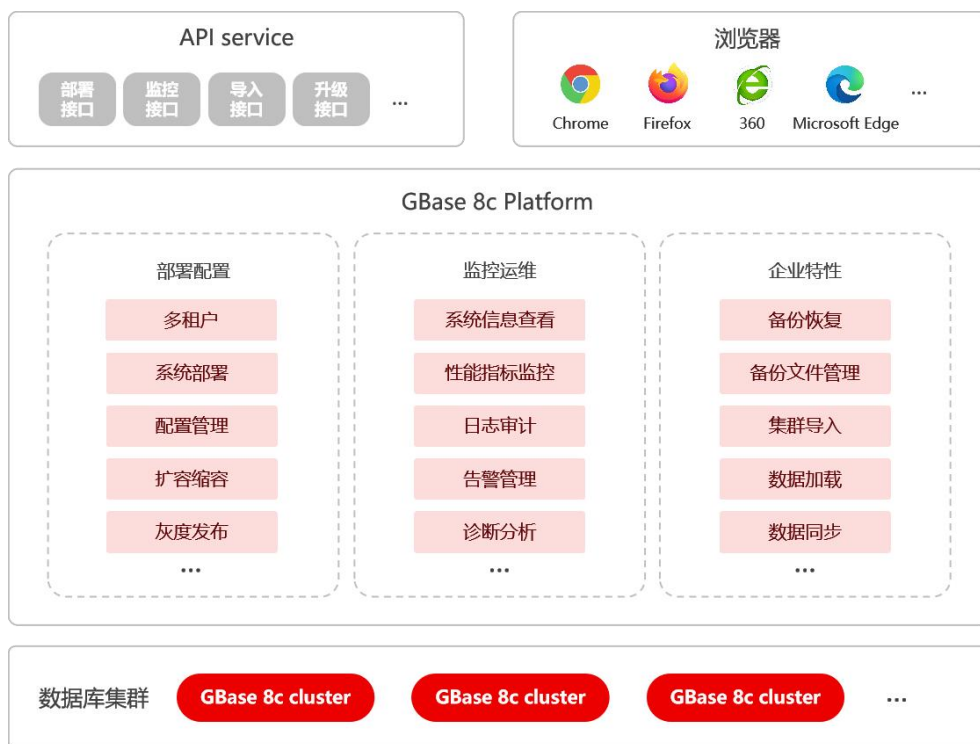
- ✓ **Coordinator 协调器**: 协调器管理和客户端的连接, 对客户端发来的 SQL 进行解析, 生成执行计划, 把执行计划发到相应的数据节点进行读写操作, 并将结果汇总返回给客户端。对于涉及跨 DN 的数据写操作, 协调者还负责协调所有参与者进行两阶

段提交。

- ✓ **Data Node 数据节点**：数据节点是数据实际存放的节点，保存表和索引等数据库对象。数据节点接收协调器下发的读写操作，并将结果返回给协调器处理。数据节点间也可以通信，进行跨数据节点的数据重分布和联合查询。
- ✓ **GTM 全局事务管理器**：全局事务管理器管理全局事务号和活动事务状态，确保系统的全局一致性。

2.2 通用管理平台

GBase 8c 通用管理平台，简称为 GBase 8c Platform，是一款功能强大的数据库集群管理平台，以图形可视化的形式，实现多租户资源隔离、系统部署、配置管理、运维监控、告警管理、备份恢复、扩容缩容、灰度发布等特性。



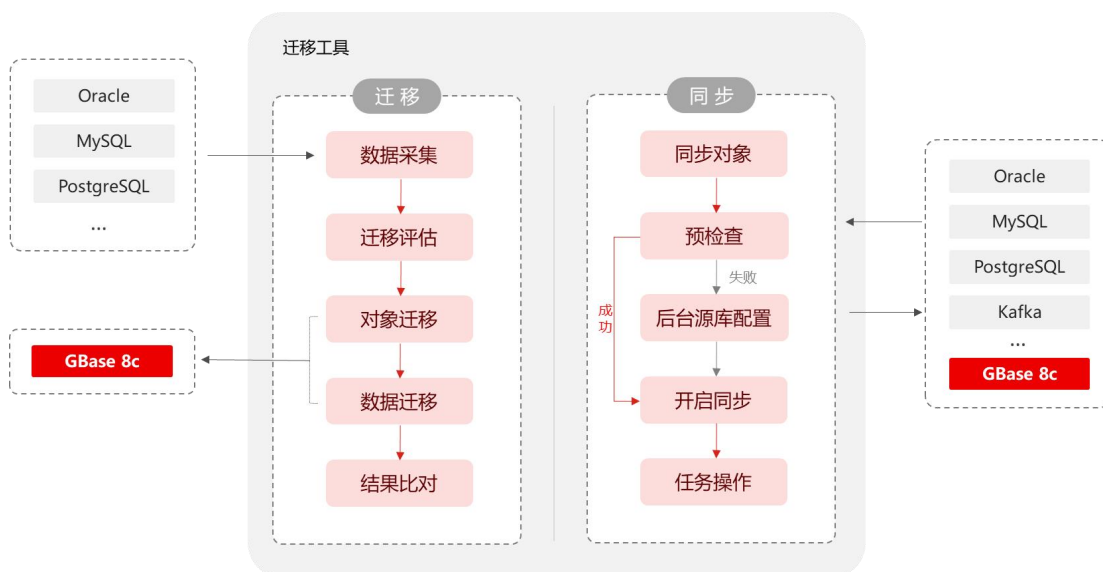
- ✓ **多租户**：支持将服务器等物理资源统筹分配给租户，并按需为租户创建、配置、管理单机数据库或数据库集群（包括主备式、高可用分布式等架构）。租户之间的资源相互隔离。
- ✓ **系统部署**：支持数据库集群的一键快速部署，并直观地显示各部署步骤的执行状态

和结果。

- ✓ **配置管理**: 支持在线参数配置管理, 通过图形化界面的方式查看集群节点参数配置项, 允许在线修改和即时生效。
- ✓ **运维监控**: 支持数据库集群状态监控和告警, 提供集群、节点、服务器等多监控维度, 深度展现各节点健康状态, 还提供定制化指标界面, 以使用户掌握集群即时或历史的运行动态。
- ✓ **性能诊断**: 支持对指定集群生成 WDR (Workload Diagnosis Report) 诊断报告, 支持诊断报告的管理操作。
- ✓ **备份恢复**: 支持以图形化界面的方式, 使用集群数据备份/恢复功能, 选择备份/恢复方式、存储介质, 并直观显示备份/恢复进度和历史记录。
- ✓ **扩容缩容**: 支持数据库集群进行在线扩容/缩容, 实现高可用组或高可用节点的横向扩展/缩减。
- ✓ **灰度发布**: 支持数据库版本的灰度发布, 通过调整负载, 实现各节点分批在线完成版本更新, 并直观显示集群升级进度和过程。

2.3 数据迁移平台

GBase 8c 数据迁移平台 (Data Migration Tool), 简称为 GBase 8c DMT, 能够实现异构数据库到 GBase 8c 数据库之间的迁移和同步, 并支持迁移评估、迁移对比等功能。



- ✓ **数据迁移**: GBase 8c DMT 支持异构数据库到 GBase 8c 数据库之间的结构迁移、全

量数据迁移和增量数据迁移。此外，在迁移过程中，还支持采集数据统计、迁移评估、迁移调整重试、迁移设置、数据级和对象级迁移对比等功能。

- ✓ **数据同步**：GBase 8c DMT 支持 GBase 8c 与 Oracle、MySQL、PostgreSQL、Kafka 等库之间的实时数据同步。用户通过创建同步任务的操作，平台后台自动搭建相应同步链路，主要支持增量数据同步的场景，支持将源端的数据变更实时同步到目标端。

3 环境及技术指标

3.1 软硬件运行环境

配置项	配置要求
服务器	x86_64、ARM 的标准 PC 服务器、PowerLinux 服务器 支持浪潮、华为、曙光、H3C 等国产服务器；
硬盘	用于安装 GBase 8c 的硬盘需最少满足如下要求： <ul style="list-style-type: none"> ● 至少 4GB 用于安装 GBase 8c 的应用程序； ● 每个主机需大约 300MB 用于元数据存储； ● 预留 70%以上且不少于 500GB 容量的磁盘剩余空间用于数据存储。 建议系统盘配置为 RAID1，数据盘配置为 RAID5，且规划 4 组 RAID5 数据盘用于安装 GBase 8c。 单台服务数据盘建议：raid5(<=8 块)；raid50(>8 块)。 有关 RAID 的配置方法请参考硬件厂商的手册或互联网上的方法进行配置，其中 Disk Cache Policy 一项需要设置为 Disabled，否则机器异常掉电后有数据丢失的风险。 GBase 8c 支持使用 SSD 盘作为数据库的主存储设备，支持 SAS、SATA、NVME 三种接口的 SSD 盘，以 RAID 的方式部署使用。
内存	功能调试建议 32GB 以上。 性能测试和商业部署时，单个节点建议 128GB 以上。 复杂的查询对内存的需求量比较高，在高并发场景下，可能出现内存不足。此时建议使用大内存的机器，或使用负载管理限制系统的并发。
存储	本地存储（SATA、SAS、NVMe 等）、阵列存储（SAN、NAS）、软件定义存储（vSAN、Ceph 等）；
CPU	功能调试最小 1×8 核 2.0GHz。 性能测试和商业部署时，单节点建议不低于 2*10 核，2.0GHz。 CPU 超线程和非超线程两种模式都支持。 说明 ：目前 GBase 8c 支持 Intel、AMD、飞腾、海光、鲲鹏等。
网络要求	支持千兆、万兆、InfiniBand，节点间数据传输建议使用万兆。 建议网卡设置为双网卡冗余 bond。有关网卡冗余 bond 的配置方法请参考硬件厂商的手册或互联网上的方法进行配置。
操作系统	CentOS、Red Hat、麒麟、统信等。

3.2 技术指标

技术指标	最大值
数据库容量	受限于服务器配置及数量
集群节点数	1024
分片数	1024

单表大小	2 ⁷⁷ B
单行数据大小	1.6 TB
每条记录单个字段的大小	1 GB
单表记录数	无限制
单表列数	250~1600 (随字段类型不同会有变化)
单表中的索引个数	无限制
复合索引包含列数	32
单表约束个数	无限制
并发连接数	10000 * CN 节点数
分区表的分区个数	32768 (范围分区) / 64 (哈希分区/列表分区)
分区表的单个分区大小	128 TB * DN 节点数
分区表的单个分区记录数	无限制

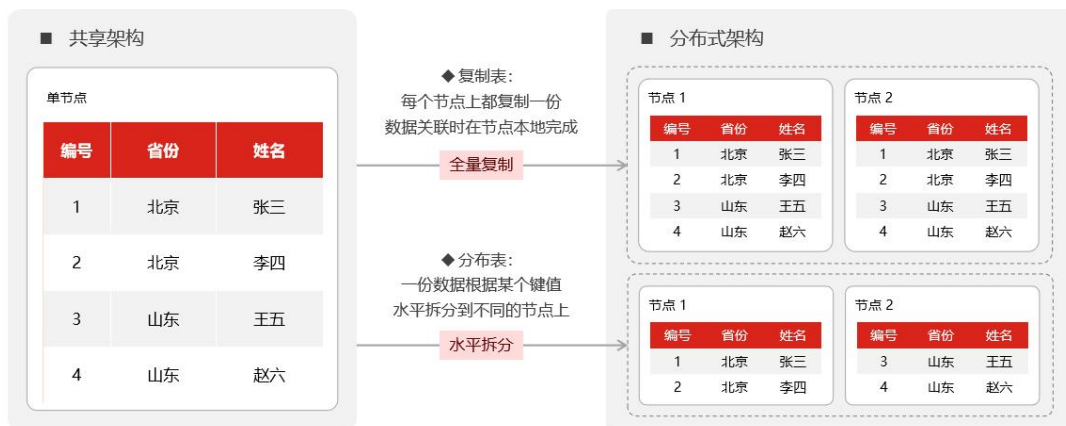
4 核心技术

4.1 自适应事务处理机制

GBase 8c 采用自适应的事务处理机制来提升系统性能。对于只需要在本地节点进行处理的事务，协调器按本地事务处理流程进行处理，不需要进行两阶段提交，以提升事务处理的效率；对于需要跨节点处理的事务，协调器协调参与者进行两阶段提交，以保障全局事务的一致性。整个事务处理的流程对客户端透明。

4.2 数据分布策略

GBase 8c 支持复制表和分布表，通过数据分布策略来避免并行计算期间的资源竞争，同时提升系统性能。复制表是指每个节点上都复制一份数据，数据关联时在节点本地完成。分布表是指一份数据根据某个键值水平拆分到不同的节点上，将单个大表拆分成若干小表，提升系统读写的性能。



复制表和分布表的适用场景如下：

表类型	操作类型	性能表现	适用场景
复制表	插入	较慢	字典表
	查询 (JOIN)	快/高并发/线性提升	小表
分布表	插入	快/高并发/线性提升	事实表
	查询 (多表单片)	快/高并发/线性提升	超大表

表类型	操作类型	性能表现	适用场景
	查询（单表多片）	较快	易分片的表
	查询（多表多片）	慢	避免出现
复制表-分布表混合	复制表对单一分布表 JOIN 查询	较快	主题表-事实表 字典表-事实表 小表-超大表

4.3 高性能

4.3.1 NUMA 优化

GBase 8c 的 NUMA 优化技术，是针对数据库在 NUMA 架构 CPU 上运行时出现的资源消耗不均、并发场景下远端 Node 时延较高等问题进行优化的技术。

通过设置执行线程与 Node 节点的亲和度,GBase 8c 数据库可以将线程绑定在指定的 CPU 核心运行，从而避免跨 Node 调度，降低远端访问时延，提升数据库对外处理性能。

同时，线程通过 Node 分配本地内存，也避免了某个 Node 节点内存耗尽而其他 Node 节点内存空闲的情况，充分使用 CPU、内存等资源，从而提升数据库对外处理性能。

4.3.2 并行技术

GBase 8c 采用并行技术来提升系统的性能和吞吐量，主要特点有：

- ✓ Coordinator 协调器制定分布式执行计划，将算子下推到数据节点，数据节点并行处理；
- ✓ 各数据节点采用多线程架构，多个线程并行处理；
- ✓ 采用 MVCC（多版本并发控制）技术，实现读写不冲突，提升读写并行处理能力；
- ✓ 支持并行查询，可以解决在复杂查询场景中，单个查询的执行时间过长造成系统并发度降低，从而影响数据库对外服务性能的问题。

4.3.3 原位更新

PostgreSQL 使用多版本并发控制 MVCC 机制：

- ✓ 当执行 delete 时, 数据库将删除元组直接标记为 dead, 并不会真正从物理上删除;
- ✓ 当执行 update 时, 数据库将会使用 unused 空间写入一个新的元组, 然后将旧元组标记为 dead, 也不进行物理删除;
- ✓ 当表上频繁 DML 时, dead tuple 会逐渐将空间耗尽, 同时做全表扫描时产生很多额外 I/O。

GBase 8c 采用原位更新技术:

- ✓ 将 new tuple 放在原位, 将 dead tuple 集中存放在 undo;
- ✓ 去除 vacuum, 保证数据回收时 IO 稳定;
- ✓ 数据空间缩减。

PG 采用追加更新方式存储数据, 也就是当修改数据时, 不是在原位置修改, 而是写入一个新记录, 这会导致空间膨胀, 也就需要定期回收过期的数据空间。这一直是 PostgreSQL 的一个弱项。

而 GBase 8c 实现了 Undo 机制, 也就可以在原位置更新数据。这带来的好处包括:

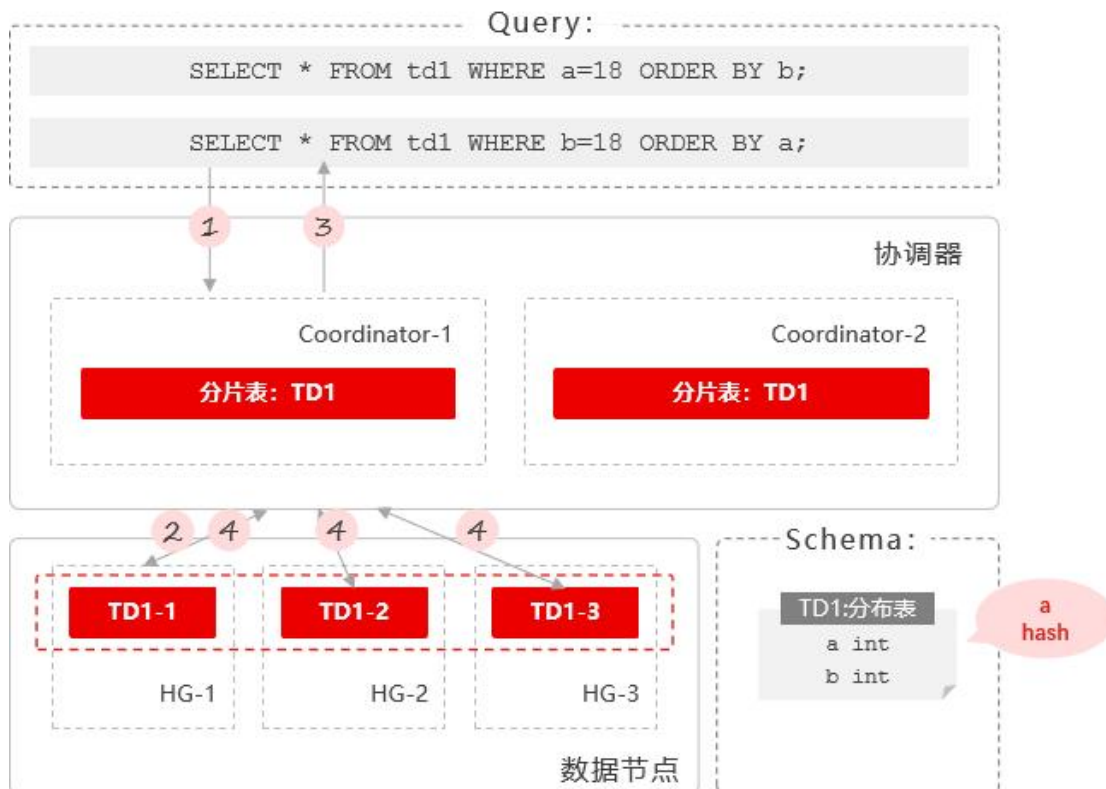
- ✓ 高性能: 对插入、更新、删除等不同负载的业务, 性能以及资源使用表现相对均衡, 相比 Append Update 引擎性能提升 10%;
- ✓ 运行平稳: 性能运行平稳, 8 小时性能滚降值从 13.8% 降低至 2.5%;
- ✓ 高效存储: 支持最大限度的原位更新, TPCC 负载下平均节约空间 15%~20%, UNDO 空间统一分配, 集中回收, 复用效率更高, 存储空间使用更加高效、平稳。

4.3.4 算子下推

算子下推是 GBase 8c 关键技术之一, 可以把各种复杂的 SQL 进行下推执行, 最小化数据移动, 这是相对于基于分库分表的中间件方案的核心优势。

4.3.4.1 单表查询下推

单表查询, 不管 SQL 的 where 条件是否带有分片键, 优化器都可以生成下推的执行计划, 包括 sort/group by 等复杂算子, 都可以下推。



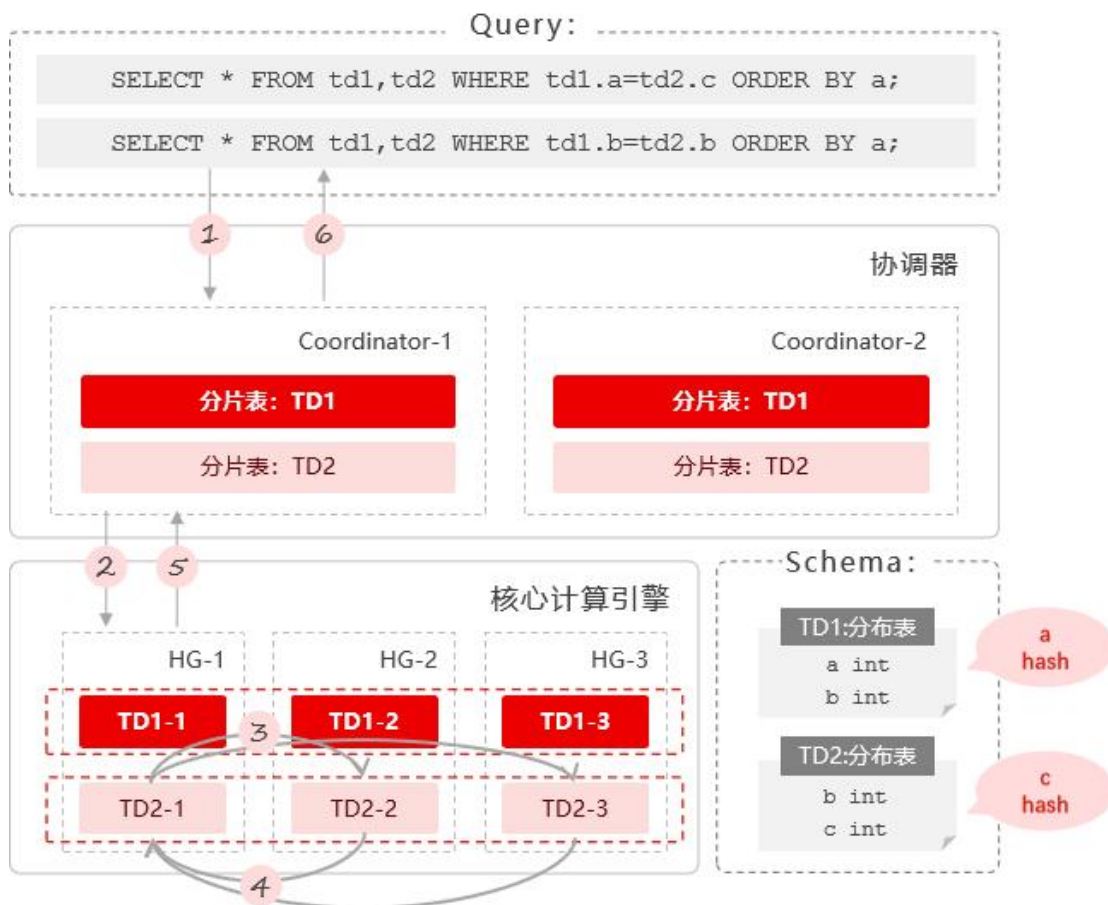
(1) 分片键上的 where 条件，直接下推到对应 DN 执行：

```
gbase=# EXPLAIN SELECT * FROM td1 WHERE a=18 ORDER BY b;
QUERY PLAN
-----
Remote Fast Query Execution (cost=0.00..0.00 rows=0 width=0)
Node/s: dn2
-> Sort (cost=38.44..38.47 rows=11 width=8)
    Sort Key: b
    -> Seq Scan on td1 (cost=0.00..38.25 rows=11 width=8)
        Filter: (a = 18)
(6 rows)
```

(2) 非分片键 where 条件:DN 先计算，CN 做结果汇总，group by 可以直接下推到 DN:

```
gbase=# EXPLAIN SELECT * FROM td1 WHERE b=18 ORDER BY b;
QUERY PLAN
-----
Remote Subquery Scan on all (dn1, dn2, dn3) (cost=0.00..1.01 rows=1 width=8)
-> Seq Scan on td1 (cost=0.00..1.01 rows=1 width=8)
    Filter: (b = 18)
(3 rows)
```

4.3.4.2 Join 查询下推



(1) 分片键上的 join 条件，直接下推到对应 DN 执行：

```
gbase=# EXPLAIN SELECT * FROM td1,td2 WHERE td1.a=td2.c ORDER BY a;
```

QUERY PLAN

```
Remote Subquery Scan on all (dn1, dn2, dn3) (cost=2.04..2.05 rows=1 width=16)
-> Sort (cost=2.04..2.05 rows=1 width=16)
    Sort Key: td1.a
    -> Nested Loop (cost=0.00..2.03 rows=1 width=16)
        Join Filter: (td1.a = td2.c)
        -> Seq Scan on td1 (cost=0.00..1.01 rows=1 width=8)
        -> Seq Scan on td2 (cost=0.00..1.01 rows=1 width=8)
(7 rows)
```

(2) 非分片键 join 条件，DN 直接做数据交换，避免 CN 成为性能瓶颈：

```
gbase=# EXPLAIN SELECT * FROM td1,td2 WHERE td1.b=td2.b ORDER BY a;
```

QUERY PLAN


```
Remote Subquery Scan on all (dn1, dn2, dn3) (cost=2.04..2.05 rows=1 width=16)
  -> Sort (cost=2.04..2.05 rows=1 width=16)
      Sort Key: td1.a
      -> Nested Loop (cost=0.00..2.03 rows=1 width=16)
          Join Filter: (td1.b = td2.b)
          -> Remote Subquery Scan on all (dn1, dn2, dn3)
(cost=100.00..101.02 rows=1 width=8)
      Distribute results by H: b
      -> Seq Scan on td1 (cost=0.00..1.01 rows=1 width=8)
      -> Materialize (cost=100.00..101.03 rows=1 width=8)
          -> Remote Subquery Scan on all (dn1, dn2, dn3)
(cost=100.00..101.02 rows=1 width=8)
      Distribute results by H: b
      -> Seq Scan on td2 (cost=0.00..1.01 rows=1
width=8)
(12 rows)
```

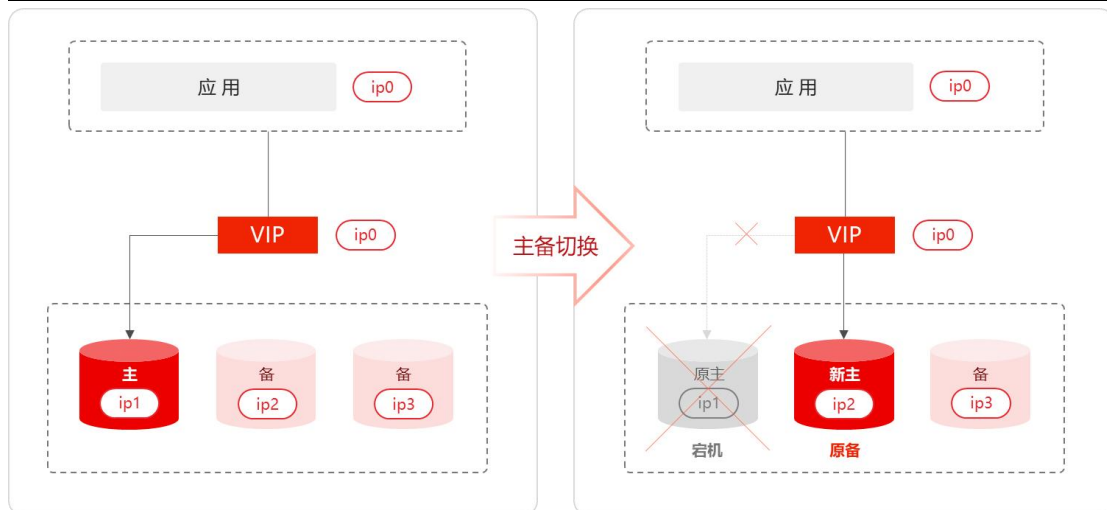
- ✓ Join 下推到 DN 执行，DN 之间直接进行数据重分布，交换数据，无需 CN 参与；CBO 优化器选择小表 t2 做重分布；
- ✓ Sort 下推到 DN，CN 只需做归并排序，避免 CN 成为性能瓶颈；

4.4 高可用

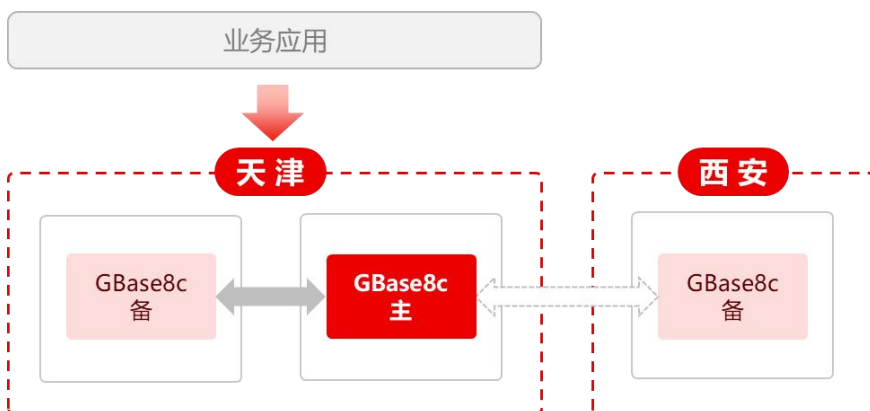
4.4.1 主备式高可用架构

GBase 8c 主备式版本支持数据多副本冗余，主备副本之间通过日志进行数据交换，保证集群任意节点故障均不影响数据库对外提供服务，数据无丢失，满足 ACID 特性。

当主节点故障（包括但不限于服务器故障、数据库服务故障、断网、断电、硬盘故障等）时，备节点可以自动升级为主，并继续对外提供服务。该过程对应用透明，不受主备 IP 地址限制，即当备机被拉起后会主动被赋予当前应用正在连接的 IP 地址，整个主备切换过程应用无感知，不影响对外服务。



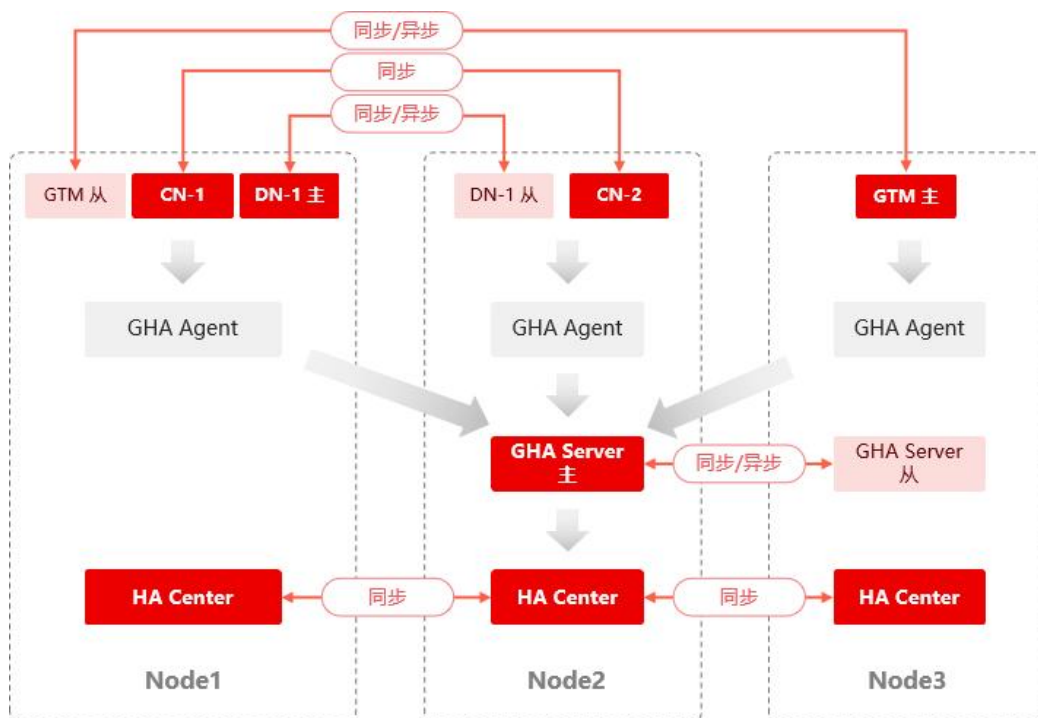
如图所示，GBase 8c 提供 VIP (Virtual IP) 功能，应用直接连接 VIP 以访问数据库主节点。当数据库发生主节点异常导致的主备切换时，无需人工切换应用连接的 IP 地址，集群会自动将 VIP 绑定至新的主节点上，以完成集群自动切换并对上层应用无感知的目的。



GBase 8c 主备式部署主备之间支持同步及异步的备份方式，一般异地灾备的节点会采用异步的备份方式，本地的备份节点会采用同步的备份方式。

4.4.2 分布式高可用架构

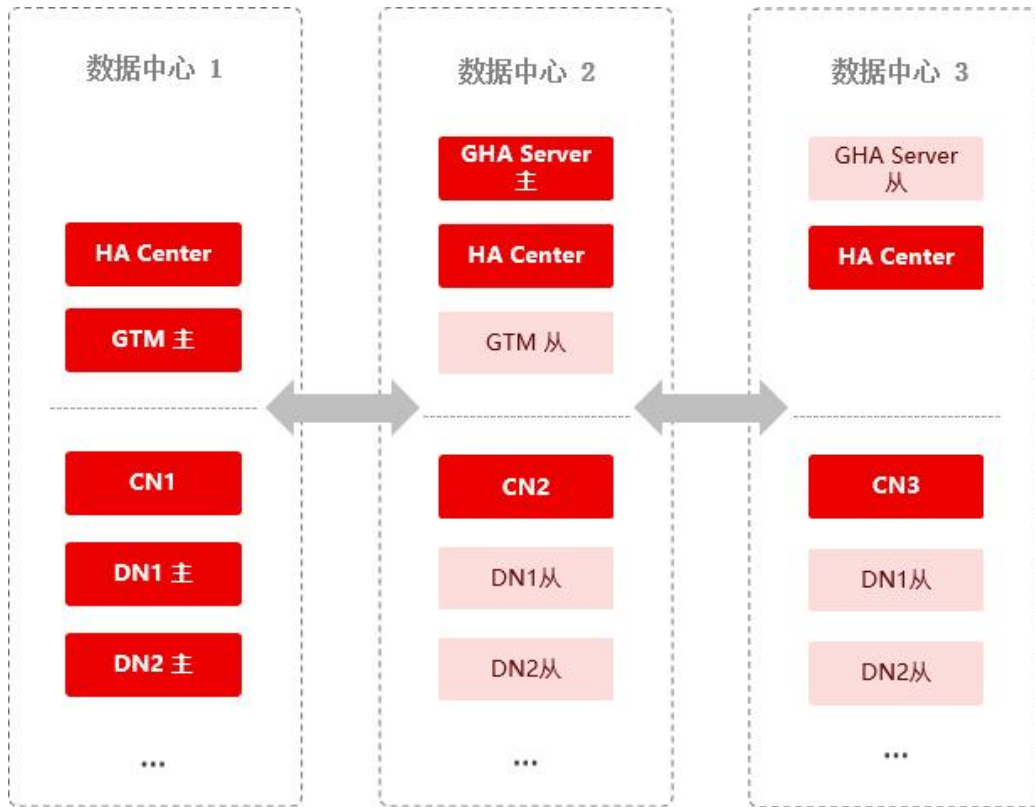
GBase 8c 高可用架构是通过分布式全组件冗余实现的，即在软件层，针对分布式集群中的每个组件，均做了组件级冗余。



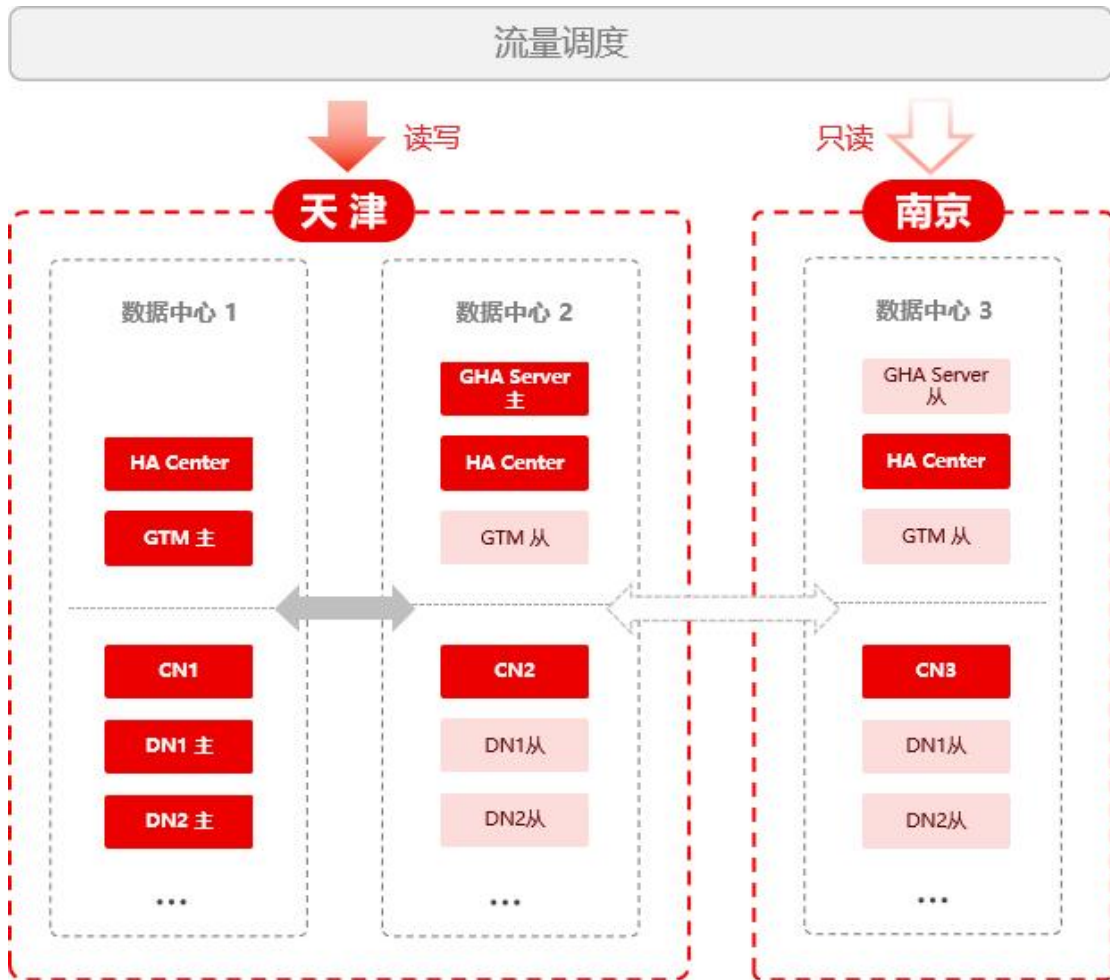
- ✓ CN: 协调器，采用完全对等的部署方式；
- ✓ DN: 数据节点，采用主备的高可用架构，主备之间可以配置同步或异步方式；
- ✓ GTM: 全局事务管理器，采用主备的高可用架构，主备之间可以配置同步或异步方式；
- ✓ HA Center: 集群状态管理器，采用 Raft 的复制协议；
- ✓ GHA Server: 集群管理器，采用主备的高可用架构，主备之间可以配置同步或异步方式；

GBase 8c 可以满足各种应用场景下对数据库不同的高可用需求。

1. 同机房容灾：采用同机房主从互备方案，可以抵御硬件级别故障，不能抵御城市级别和机房级别灾难。故障自动切换，RPO=0，RTO 秒级。
2. 同城容灾：采用同城主从互备方案，可以抵御硬件级别故障和机房级别灾难，不能抵御城市级别灾难。故障自动切换，RPO=0，RTO 秒级。同城容灾需要其中两机房之间距离小于 50 千米。



3. 异地灾备：GBase 8c 集群不同节点采用各自对应的高可用部署方式，两地间采用异步复制的备份方式。可以抵御硬件级别故障和机房级别、城市级别灾难，两地之间距离可以大于 1000 千米。



4. 异地多活：GBase 8c 支持多集群部署，集群间数据双向同步，数据以某一维度进行分区调度。可以抵御硬件级别故障和机房级别、城市级别灾难，两地之间距离可以大于 1000 千米。



4.5 分布式事务

GBase 8c 通过 GTM 全局事务管理器和本地两阶段提交技术，提供分布式强一致事务的能力，同时，对于追求性能的新兴数据库业务，也支持可选的最终一致性事务的能力。

分布式事务原子性和两阶段提交协议

为了保证分布式事务的原子性，防止出现部分 DN 提交、部分 DN 回滚的“中间态”事务，GBase 8c 采用两阶段提交（2PC）过程，实现跨节点分布式事务。

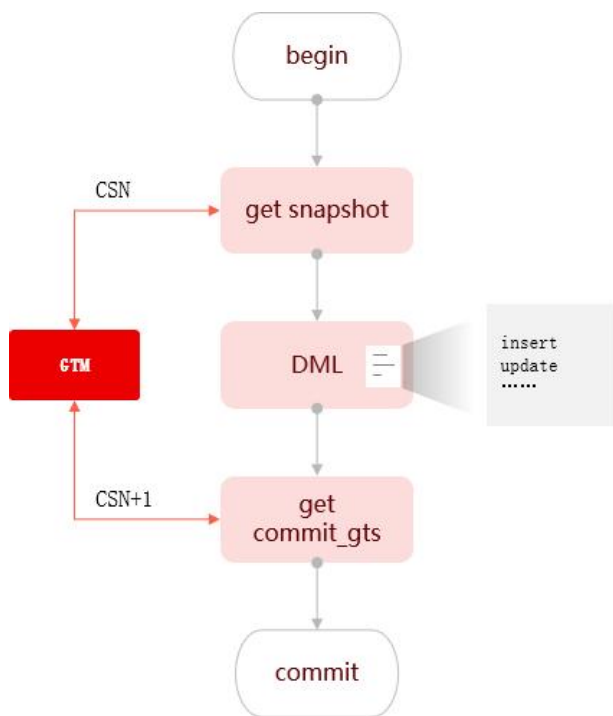
一、准备阶段（prepare phase），在这个阶段，将所有提交操作所需要使用到的信息和资源全部写入磁盘，完成持久化；

二、提交阶段（commit phase），根据之前准备好的提交信息和资源，执行提交或回滚操作。

一旦准备阶段执行成功，那么提交需要的所有信息都完成持久化落盘，即使后续提交阶段某个 DN 发生执行错误，该 DN 可以再次从持久化的提交信息中尝试提交，直至提交成功。最终该分布式事务在所有 DN 上的状态一定是相同的，要么所有 DN 都提交，要么所有 DN 都回滚。因此，对外来说，该事务的状态变化是原子的。

分布式事务一致性和全局事务管理

GBase 8c 采用了基于全局事务提交时间戳的 TSO 方案，保证分布式事务一致性。处理流程如下图所示：



1. GTM 负责维护全局时间戳 CSN；
2. 当开启事务时从 GTM 获取当前的时间戳
3. 当事务提交的时候重新获取一遍时间戳 (CSN+1)

这种方式的优势在于：

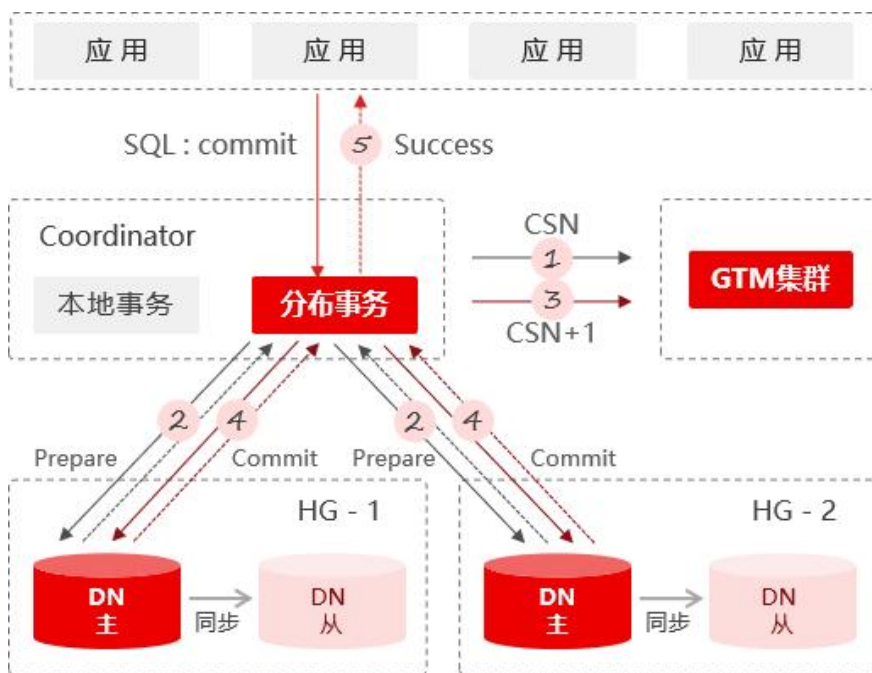
- ✓ 使用全局逻辑时间戳 CSN 号替代传统的活跃事务列表作为全局快照，可以降低所有节点到事务管理器 GTM 节点的网络开销，使得全局事务管理节点不再容易成为分布式事务处理的瓶颈节点；
- ✓ 使用逻辑时间戳 CSN 号本质上是为所有的写事务在全局进行内部排序，为处理数据库双相同步等方面提供了便捷。

示例：

```
Begin; //Transaction Start
```

1. Select * from t1 where id = 1; //单节点查询

2. `Select * from t1, t2 where t1.id = t2.id;`//跨节点查询
 3. `insert into t1 values(1, "aaa");` //单节点写入
 4. `Update t1 set name = " bbb" where id = 1;` // 单节点更新
 5. `Delete from t2 where id < 10;` //跨节点删除
- `Commit;` //Transaction Commit



4.6 全局死锁解除

全局死锁：数据库集群内，多个 CN、DN 上的多个数据库进程间互相调用资源而出现循环等待的情况，称为全局死锁。

例如：

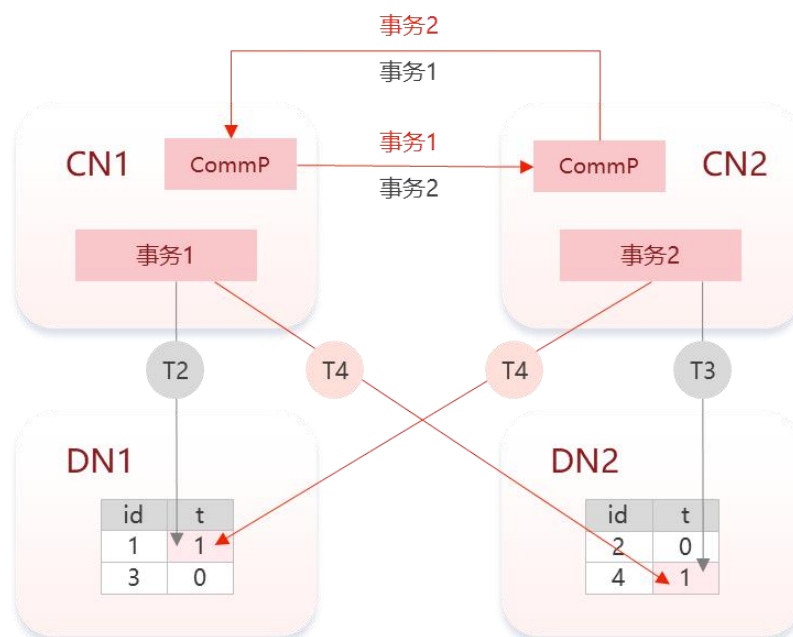


在 T1 时刻，事务一 Begin;

在 T2 时刻，事务一 update id=1 的 t 值，同时事务二 Begin;

在 T3 时刻，事务二 update id=4 的 t 值;

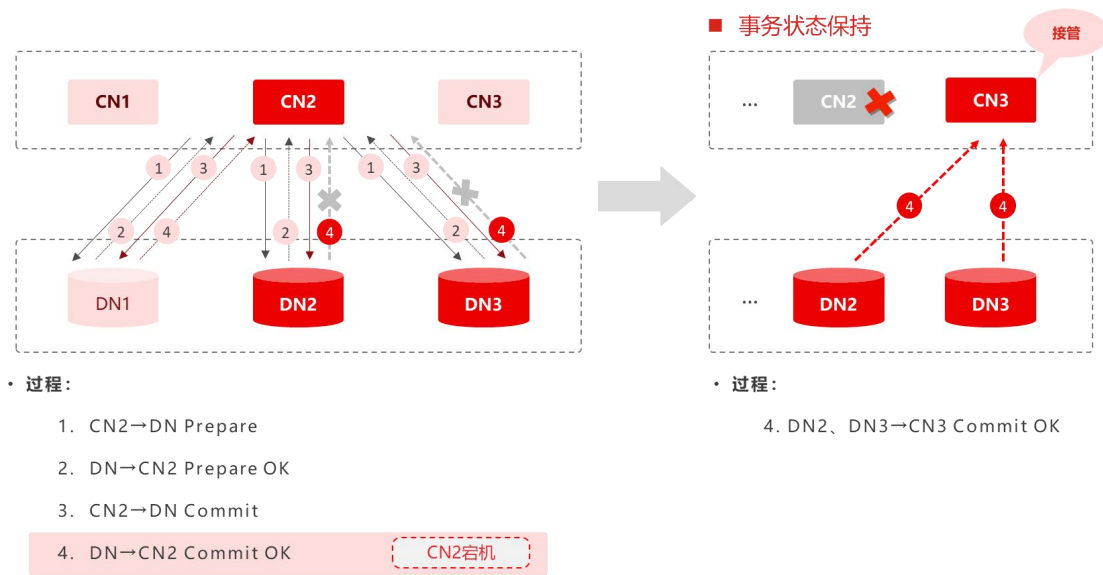
在 T4 时刻，事务一要更新 id=4 的 t 值，同时事务二要更新 id=1 的 t 值，此时两事务出现循环等待，就发生了全局死锁的情况。



而在 GBase 8c 数据库中，节点间检测出死锁环之后，将首个发现死锁环节点的事务退出的操作，从而解决全局死锁的问题。

4.7 事务状态保持

GBase 8c 具备事务状态保持能力，任意协调器节点 (CN) 宕机后，都不影响该节点正在进行的事务状态，事务会自动迁移到其它 CN 上，继续顺利运行，确保数据库处理能力不会中断。



上图中 CN3 节点接管事务后，无需重复前面已经成功提交的事务状态，而是继续完成宕机的 CN2 节点未完成的状态来完成本次事务。整个过程对上层业务无感知，数据库集群内任意节点宕机均不会造成死锁或异常等待情况。

4.8 全局 CDC

GBase 8c 支持全局 CDC (Change Data Capture, 变化数据捕获) 特性，方便用户进行数据库的全局备份和数据抽取。CDC 变化数据捕获的方式主要包括时间戳、快照、触发器和日志，其中 GBase 8c 基于日志的 CDC 方式，对源数据库的性能不会产生影响。

4.9 备份恢复

海量的业务数据不仅仅给数据处理和分析查询的性能带来挑战，对数据备份和恢复的要

求也更高。因为数据量巨大，如果没有高效的备份和恢复能力，在意外、故障或灾难发生时，无法及时使数据库得到恢复，系统和业务的可用性就无法得到保障。

GBase 8c 集群具有全局备份和恢复的能力，支持全量备份、恢复，支持增量备份、恢复。在通用管理平台上，可以进行备份方式和备份时间频率的配置，并且能够查看全部的备份记录。提供全面的基于集群级、库级、表级的备份和恢复功能，包括：

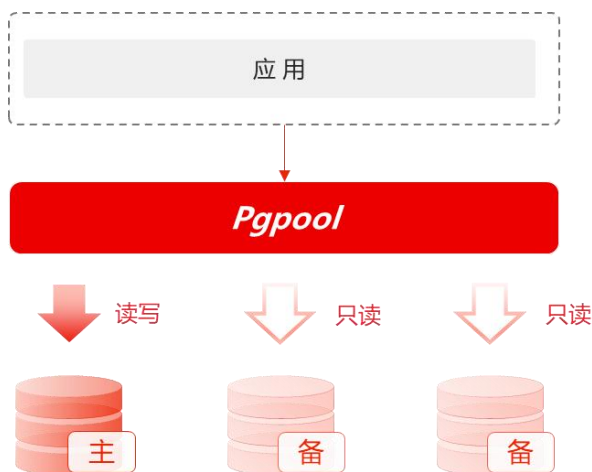
1. 全量备份和恢复；
2. 增量备份和恢复：允许基于任意一个备份点进行数据恢复。

支持并行恢复

主机日志传输到备机时，备机日志落盘的同时，发送给重做恢复分发线程，分发线程根据日志类型和日志操作的数据页发给多个并行恢复线程进行日志重做，保证备机的重做速度跟上主机日志的产生速度。这样备机实时处于 ready 状态，从而实现瞬间故障切换。

4.10 读写分离

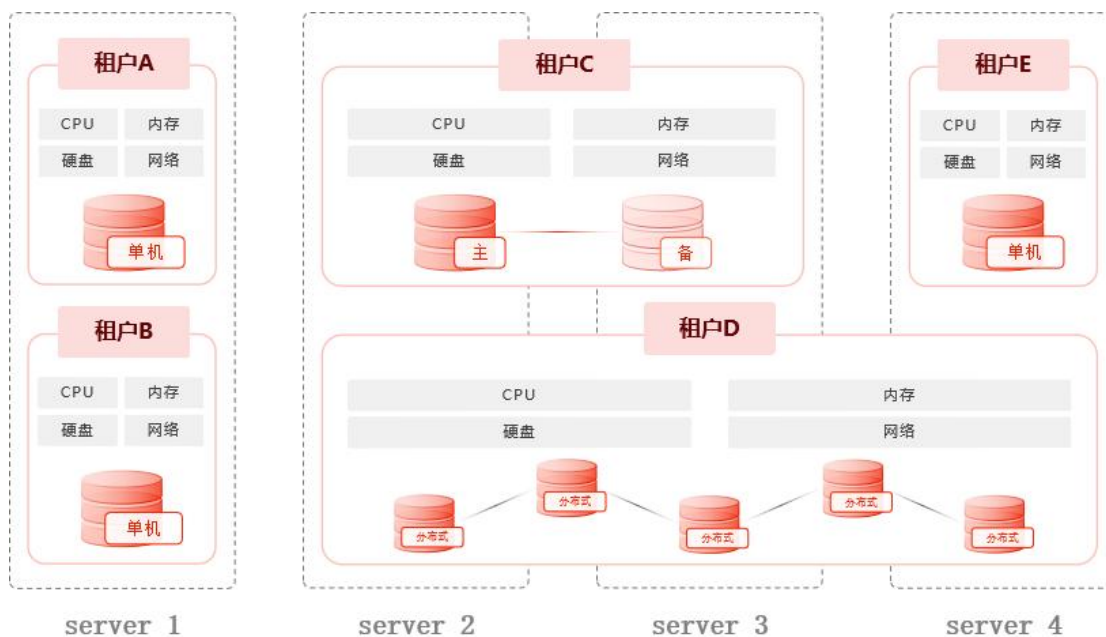
GBase 8c 支持语句级及连接级读写分离。其中，语句级读写分离，可以自动识别读、写请求，并分别分配至主、备节点；连接级读写分离支持通过配置连接的方式，将读写连接配置在主节点，将只读连接配置在备节点。



5 产品高级特性

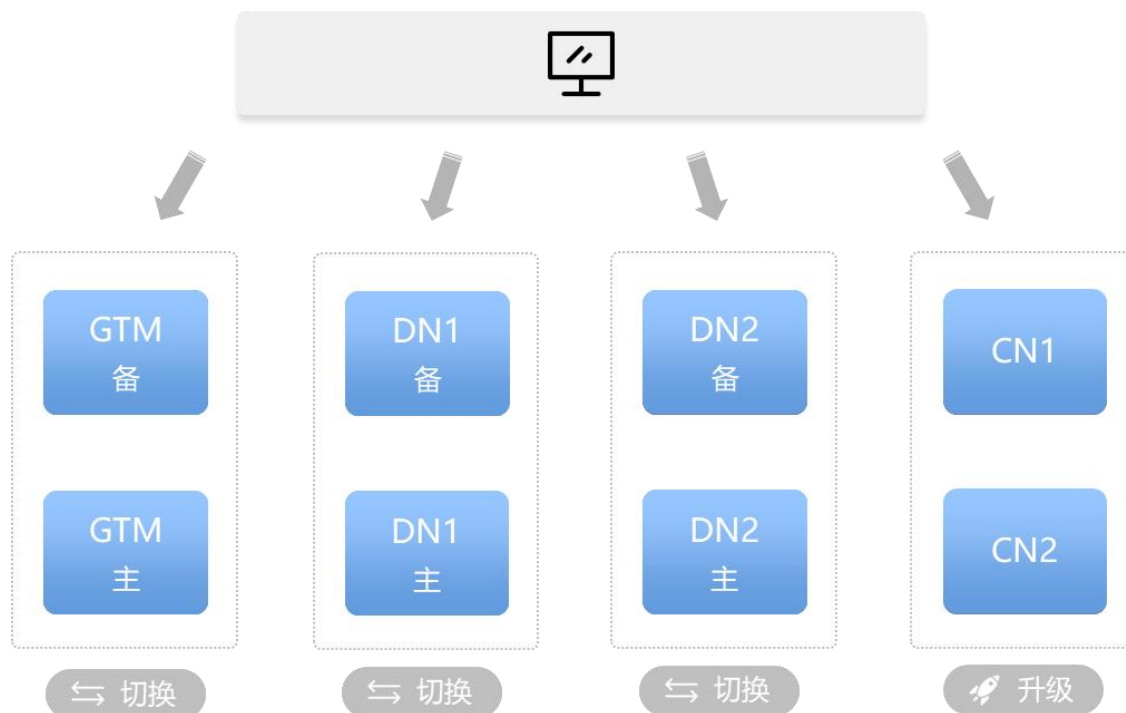
5.1 多租户

GBase 8c 支持基于多租户的资源隔离能力，不同租户之间资源相互隔离，并且可以设置每个租户允许使用的资源。每个租户可自由选择单机部署模式、主备部署模式以及分布式部署模式。GBase 8c 多租户的设置有效隔离了 CPU 资源、内存资源、硬盘资源、网络资源，提供了有力的混合负载管理能力。



5.2 灰度发布

GBase 8c 支持灰度发布，可以在保证业务持续可用的条件下，进行数据库版本升级和发布。



在灰度发布的过程中，GBase 8c 可以秒级切换，具备系统持续可用的能力。

5.3 强一致性分布式事务

GBase 8c 通过两阶段提交来实现分布式事务，确保所有跨节点事务的原子性、一致性、隔离性和持久性。两阶段提交可以保证分布式事务要么全部提交成功，要么全部提交失败，不会产生事务在部分节点提交成功，部分节点提交失败的情况。

对于提交成功的分布式事务，GBase 8c 确保从任意节点访问都可以获得同样的结果，实现分布式事务的强一致性。

5.4 多部署方式

随着虚拟化和云计算技术的不断深入应用，业务上云成了 IT 架构发展的趋势，进而对数据库的云上部署能力提出了要求。

GBase 8c 支持物理服务器部署、虚拟机部署、容器部署、云上裸金属服务器部署、私有云部署、公有云部署等多种部署方式，为不同 IT 架构下的数据库部署提供了多种选择。

1. 物理服务器部署：传统的部署方式，适合对数据库资源需要物理隔离，IT 架构尚未云化的客户。

2. 虚拟机/容器部署：对于希望采用虚拟化部署，但 IT 架构尚未云化的客户可以使用容器来部署 GBase 8c。

3. 云上裸金属部署：适合 IT 架构已经云化，同时数据库资源需要物理隔离的客户，可以将 GBase 8c 部署在云上裸金属服务器。

4. 私有云/公有云部署：对于 IT 架构已经云化，不需要数据库资源物理隔离的客户，可以选择将 GBase 8c 部署在私有云或者公有云的云主机。

5.5 安全特性

GBase 8c 具有完备的安全特性，包括身份认证和鉴别、数据透明加密、三权分立、安全审计等。

1. 身份认证和鉴别：在 GBase 8c 中，每个数据库用户都有一个不可重复的唯一性用户标识，并在数据库整个生命周期实现该用户标识的唯一性。用户在使用数据库时必须首先给出用户标识，通过检验合格后才能进入使用。数据库用户的密码使用加密算法处理后存储在 GBase 8c 的系统表中，密码加密后不可逆，从而保证了密码自身的安全性。

2. 数据透明加密：GBase 8c 的数据存储加密采用库内加密的方式，在数据库内核存储引擎中进行数据加解密处理，对于用户来说是完全透明的。GBase 8c 支持库、表、字段等不同数据粒度的存储加密，不同数据对象可采用不同的加密密钥，如一表一密，可有效防止单点突破，保证更高的数据安全性。

3. 安全审计：GBase 8c 具有独立的审计系统，它能定义相关的审计事件，记录用户的相关操作，并能记录用户标识、身份鉴别等的审计数据，能进行相关的审计分析并自动报警，并支持对审计数据进行查阅。为便于独立审计，保证更高的系统安全性，GBase 8c 设有专门的安全审计员进行审计管理，安全审计员可利用专门的审计操作界面对审计事件作选择，查阅有关审计数据，处理报警信息。

4. 三权分立：GBase 8c 把数据库的用户由原来单一的超级数据库管理员变成三类角色：安全管理员、审计管理员、数据管理员。它们分别承担着不同的职责，三者之一不涉及其它两者的权力范围，从而实现整个数据库系统的分权管理，即三权分立原则。三类用户的具体

职责分配大致如下：安全管理员主要负责完成系统的安全管理功能，审计管理员负责完成系统的审计功能，数据管理员主要负责完成存取控制、系统维护管理等功能。这三类管理员用户之间分工明确，各司其责，既相互制约又相互配合，共同实现数据库的安全管理功能。

5. 数据动态脱敏：为了在一定程度上限制非授权用户对隐私数据的窥探，可以利用动态数据脱敏（Dynamic Data Masking）特性保护用户隐私数据。在非授权用户访问配置了动态数据脱敏策略的数据时，数据库将返回脱敏后的数据而达到对隐私数据保护的目的。在不改变源数据的前提下，通过在脱敏策略上配置针对的用户场景（FILTER）、指定的敏感列标签（LABEL）和对应的脱敏方式（MASKING FUNCTION）来灵活地进行隐私数据保护。

5.6 空间数据库

GBase 8c 具有空间数据库功能，GBase 8c 为 GIS 提供与应用相关的地理空间数据的存取功能，提供地理信息的快速检索功能。

GBase 8c 支持地理坐标系和投影坐标系，支持不同坐标系之间的相互转换；支持 OGC“Simple Features for SQL”规范中指定的对象和函数，同时也支持“SQL/MM”规范的几何对象；支持主流地理空间数据类型（矢量、栅格、网格、影像、网络、拓扑）；支持三种基本的几何形状（点、线、面）；支持空间索引：R-tree、函数索引、空间索引分区。

6 开发接口

GBase 8c 接口驱动可以有效实现对上层应用请求的负载均衡。

6.1 ODBC

GBase 8c ODBC 是 GBase 8c 的 ODBC 驱动程序，它提供了访问 GBase 8c 的所有 ODBC 功能。GBase 8c ODBC 支持 ODBC 3.5X 一级规范（全部 API+2 级特性）。用户可以通过 ODBC 数据源管理器调用 GBase 8c ODBC 驱动访问 GBase 8c 数据库，或者直接调用 GBase 8c ODBC 驱动访问 GBase 8c 数据库。另外，可视化编程工具如 C++ Builder、Visual Studio 等也可以利用 GBase 8c ODBC 访问 GBase 8c 数据库。

ODBC 驱动管理器是管理 ODBC 应用和驱动程序之间通信的库，它的主要功能包括：

1. 解析数据源名字 (DSN) ；
2. 装载和卸载驱动程序；
3. 处理 ODBC 函数调用或传递它们到驱动程序。

6.2 JDBC

JDBC (Java Database Connectivity, java 数据库连接) 是一款用于执行 SQL 语句的 Java API, 可以为多种关系数据库提供统一访问接口, 应用程序可基于它操作数据。GBase 8c 提供了对 JDBC 4.0 特性的支持, 需要使用 JDK1.6 及以上版本编译程序代码, 不支持 JDBC 桥接 ODBC 方式。

6.3 ADO.NET

GBase 8c ADO.NET 是一款提供 .NET 应用程序与 GBase 8c 数据库之间方便、高效、安全交互的接口程序, 开发人员可以使用任何一种 .NET 开发语言 (C#、VB.NET、F#) 通过 GBase 8c ADO.NET 操作 GBase 数据库。

GBase 8c ADO.NET 支持以下特性：

1. 支持集群负载均衡功能；

-
2. 支持 GBase 数据库全部特性，如：视图等；
 3. 支持 Windows 平台下的 TCP/IP 套接字连接；
 4. 支持 Linux 平台下的 TCP/IP 套接字或 Linux 套接字连接；
 5. 无需安装 GBase 数据库的客户端，可通过 GBase 8c ADO.NET 类库实现完整的管理功能。

6.4 C API

GBase 8c C API 是 GBase 8c 数据库提供的 C 语言访问库。应用可以通过调用 GBase 8c C API 访问 GBase 8c 数据库。GBase C API 提供了如下功能：

1. 创建和断开客户端与服务器的连接；
2. 直接执行 SQL 语句；
3. 获取执行 SQL 的结果集；
4. 获取错误信息。

6.5 Python API

GBase 8c Python API 是 Python 语言连接并使用 GBase 8c 数据库的接口驱动程序。GBase Python API 基于 Python Database API Specification 标准编写。

接口兼容标准的同时并支持如下特性：

1. 支持 Python 2.x 和 Python 3.x；
2. 完全支持 GBase 8c 的特性；
3. 完全支持 SQL 标准语法；
4. 支持二进制流插入、更新；
5. 支持批量插入优化；
6. 支持多 SQL 语句执行和获取多结果集；
7. 支持 TCP/IP 协议。

GBASE[®]

南大通用数据技术股份有限公司
General Data Technology Co., Ltd.



微信二维码



■ ■ 技术支持热线：400-013-9696